



دانشکده علوم ریاضی

اصول پردازش تصویر

نیم سال اول ۱۳۹۹-۱۴۰۰

مدرس: دکتر مصطفی کمالی تبریزی

تمرین سری پنجم – سوال دوم

شماره دانشجویی: ۹۷۱۰۰۳۹۸

نام و نام خانوادگی: سیدعلیرضا خادم

زمان محدودی اجرا: کمتر از ۵ ثانیه

ابعاد تصویر apple: $512px \times 512px$

ابعاد تصویر orange: $512px \times 512px$

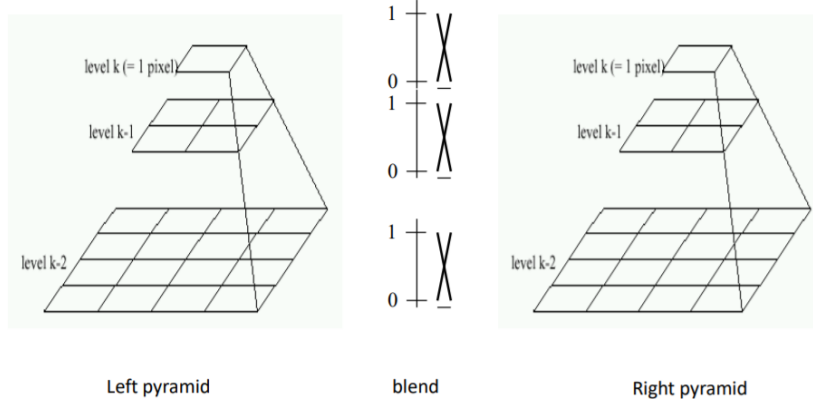
موارد لازم.

برای اجرا لازم است تا تصاویر apple.jpg و orange.jpg در مسیر EX5_Q2 قرار داشته باشد. همچنین در پیاده سازی این سوال از کتابخانه های cv2 و numpy استفاده شده است که قبل از اجرا بایستی این کتابخانه ها روی سیستم شما نصب باشد.

روند کلی حل.

ایده کلی حل این سوال استفاده از هرم های gaussian و laplacian برای blendig است.

- At low frequencies, blend slowly
- At high frequencies, blend quickly



در این روش ابتدا برای دو تصویر laplacian pyramid را به دست می آوریم، و برای mask هم gaussian

pyramid را به دست می‌آوریم. در ادامه هر سطح از هرم را با استفاده از سطح متناظر mask از gaussian pyramid آن ترکیب می‌کنیم. (با توجه به رابطه‌ای که در ادامه آمده است).

$$L_{12}^i = L_1^i \cdot R^i + L_2^i \cdot (1 - R^i)$$

Image 1 at level i of Laplacian pyramid Region mask at level i of Gaussian pyramid

Pointwise multiply

کلیت پیاده‌سازی الگوریتم هم در ادامه آمده است.

Laplacian Pyramid Blending

Implementation:

1. Build Laplacian pyramids for each image
2. Build a Gaussian pyramid of region mask
3. Blend each level of pyramid using region mask from the same level

$$L_{12}^i = L_1^i \cdot R^i + L_2^i \cdot (1 - R^i)$$

Image 1 at level i of Laplacian pyramid Region mask at level i of Gaussian pyramid

Pointwise multiply

4. Collapse the pyramid to get the final blended image

Burt and Adelson 1983

توضیح کد.

برنامه در مجموع حاوی ۲ فایل با فرمت .py می‌باشد که توضیحات هر فایل در پایین آمده است.

`get_gaussian_pyramid(src_image, num_levels)`

این تابع یک تصویر و یک عدد را به عنوان ورودی می‌گیرد و هرم gaussian مربوط به `src_image` با `num_levels` سطح خروجی می‌دهد.

`get_laplacian_pyramid(src_image, num_levels)`

این تابع یک تصویر و یک عدد را به عنوان ورودی می‌گیرد و هرم laplacian مربوط به `src_image` با `num_levels` سطح خروجی می‌دهد.

`get_blended_pyramid(lp_source, lp_target, gp_mask)`

این تابع هرم laplacian تصویر `source`، هرم laplacian تصویر `target` و هرم gaussian تصویر `mask` را به عنوان ورودی می‌گیرد و `blended_pyramid` را با توجه به روابط زیر ساخته به عنوان خروجی برمی‌گرداند.

$$L_{12}^i = L_1^i \cdot R^i + L_2^i \cdot (1 - R^i)$$

Diagram illustrating the formula for L_{12}^i . The formula is $L_{12}^i = L_1^i \cdot R^i + L_2^i \cdot (1 - R^i)$. Arrows point from the terms to their descriptions: L_1^i is "Image 1 at level i of Laplacian pyramid", R^i is "Region mask at level i of Gaussian pyramid", and the operation \cdot is labeled "Pointwise multiply".

`reconstruct_channel(blended_pyramid)`

این تابع `blended_pyramid` را به عنوان ورودی می‌گیرد و با استفاده از تابع `cv2.pyrUP` عملیات collapse را انجام می‌دهد.

`blend_channel(source_channel, target_channel, mask)`

این تابع یک کانال از تصویر `source` و یک کانال از تصویر `target` را به همراه `mask` به عنوان ورودی می‌گیرد و با استفاده از توابع `get_blended_pyramid` و `reconstruct_channel` این دو چنل را با هم ترکیب کرده و نتیجه را به عنوان خروجی برمی‌گرداند.

`pyramid_blend(source, target, mask)`

این تابع تصاویر `source` ، `target` و `mask` را به عنوان ورودی می‌گیرد و با استفاده از تابع `blend_channel` برای هر یک از کانال‌های تصاویر، به صورت مجزا `blending` را انجام می‌دهد و تصویر حاصل را به عنوان خروجی برمی‌گرداند.

`mouse_handler`

این تابع رویدادهای مربوط به `mouse` را مدیریت می‌کند تا کاربر بتواند رئوس چندوجهی `mask` را ورودی دهد.

`get_mask(shape, vertices)`

این تابع رئوس و ابعاد `mask` را ورودی می‌گیرد و با استفاده از تابع `fillConvexPoly` ، `mask` را می‌سازد و به عنوان خروجی برمی‌گرداند.

○ `q2.py`

در این فایل ابتدا تصاویر را لود می‌کند و در ادامه تصویر `apple_image` برای کاربر نمایش داده می‌شود تا راس‌های چندوجهی `mask` را با کلیک چپ روی تصویر مشخص کند. منتظر می‌ماند تا کاربر بعد از وارد کردن راس‌ها، دکمه `ESC` از کیبورد را بفشارد. بعد از فشردن دکمه `ECS` توسط کاربر، با توجه به رئوسی که کاربر وارد کرده `mask` ساخته می‌شود. در نهایت با فراخوانی تابع `pyramid_blend` روی ورودی‌هایی که مشاهده می‌کنید، `result` را محاسبه می‌کند و با نام `res2.jpg` در مسیر `EX5_Q2/results` ذخیره می‌کند.